



درس طراحی زبان‌های برنامه‌سازی

دکتر محمد ایزدی

کوییز دوم

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

نیم سال دوم ۱۴۰۰-۱۳۹۹

مهلت زمان امتحان:

۵۵ دقیقه



به موارد زیر توجه کنید:

- * مدت زمان امتحان ۵۵ دقیقه است. پس از ۵۵ دقیقه امکان ارسال جواب در کورا وجود نخواهد داشت.
- * سوالات کدی باید به زبان Racket نوشته شوند.
- * جواب تمام سوال‌ها را به صورت دست‌نویست و یا تایپی بنویسید. سپس تمام جواب‌ها را در یک فایل pdf با نام دلخواه گذاشته و در قسمت مربوطه در کورا آپلود کنید. دقت کنید که در کورا تنها امکان ارسال pdf وجود دارد.
- * تنها ارسال نهایی شما در کورا تصحیح خواهد شد. دقت کنید که در صورت ارسال چند جواب، کورا به صورت خودکار آخرین ارسال را به عنوان ارسال نهایی در نظر می‌گیرد.
- * لطفا جواب سوال‌ها را از یکدیگر کپی نکنید. در صورت وقوع چنین مواردی مطابق با سیاست درس رفتار می‌شود.



سوال ۱

همانطور که می دانید اعداد صحیح مثبت را می توان به روش های مختلفی ذخیره کرد. در این سوال میخواهیم ذخیره ی اعداد صحیح در روش Bignum را پیاده سازی کنیم. در این روش ابتدا یک پایه در نظر گرفته می شود (مثلا N) و سپس هر عدد را به شکل لیستی از اعداد بین 0 و $N - 1$ ذخیره می کنیم. عضو i ام از این لیست نمایانگر ضریب N^i در تجزیه ی عدد به پایه ی N است. (می توانید تصور کنید که یک عدد را به پایه ی N می برید و ضرایب را ذخیره می کنید.) توابع *zero*، *successor*، و *predecessor* را برای این نوع نمایش اعداد پیاده سازی کنید. فرض کنید در تعریف *zero* عدد N به عنوان ورودی داده می شود.

سوال ۲

نوع داده ی گراف ساده را در نظر بگیرید. برای هر یک از توابع زیر یک توصیف^۱ نوشته و سپس این توابع را پیاده سازی کنید.

- empty-graph
ورودی نمی گیرد و یک گراف خالی (بدون راس و یال) برمی گرداند.
- add-vertex
یک گراف به همراه یک عدد صحیح مثبت می گیرد و یک راس به نام آن عدد به گراف اضافه کرده و گراف نهایی را برمی گرداند. اگر راس تکراری بود، گراف ورودی را برمی گرداند.
- add-edge
یک گراف به همراه دو عدد به عنوان دو راس موجود می گیرد و یک یال بین آن دو راس اضافه می کند. گراف نهایی را برمی گرداند. در صورت ناموجود بودن یکی از دو راس ورودی (یا هر دو) یا موجود بودن یال بین آن دو راس، گراف ورودی را برمی گرداند.
- remove-edge
یک گراف به همراه دو عدد به عنوان دو راس موجود می گیرد، یال بین آن دو راس را حذف می کند و گراف نهایی را برمی گرداند. در صورت ناموجود بودن یکی از دو راس ورودی (یا هر دو) یا ناموجود بودن یال بین آن دو راس، گراف ورودی را برمی گرداند.

^۱specification



- adjacent?

یک گراف به همراه دو عدد به عنوان دو راس موجود می‌گیرد و چک می‌کند آیا بین آن دو راس یال وجود دارد یا نه. در نهایت یک boolean برمی‌گرداند. در صورتی که یکی از دو راس ورودی (یا هر دو) ناموجود بودند، false برگردانده شود.

در صورت نیاز به جزییات بیشتر در پیاده‌سازی هریک از موارد بالا، در پاسخنامه فرضیات خود را بنویسید.

موفق باشید